applied sciences

MDPI

Article

An Approach to the Creation and Presentation of Reference Gesture Datasets, for the Preservation of Traditional Crafts

Nikolaos Partarakis, Xenophon Zabulis *, Antonis Chatziantoniou, Nikolaos Patsiouras and Ilia Adami

Institute of Computer Science, Foundation for Research and Technology (ICS-FORTH), N. Plastira 100, Vassilika Vouton, GR-700 13 Heraklion, Crete, Greece; partarak@ics.forth.gr (N.P.); hatjiant@ics.forth.gr (A.C.); patsiouras@ics.forth.gr (N.P.); iadami@ics.forth.gr (I.A.)

* Correspondence: zabulis@ics.forth.gr; Tel.: +30-281-039-1696

Received: 31 August 2020; Accepted: 15 October 2020; Published: 19 October 2020

Abstract: A wide spectrum of digital data are becoming available to researchers and industries interested in the recording, documentation, recognition, and reproduction of human activities. In this work, we propose an approach for understanding and articulating human motion recordings into multimodal datasets and VR demonstrations of actions and activities relevant to traditional crafts. To implement the proposed approach, we introduce Animation Studio (AnimIO) that enables visualisation, editing, and semantic annotation of pertinent data. AnimIO is compatible with recordings acquired by Motion Capture (MoCap) and Computer Vision. Using AnimIO, the operator can isolate segments from multiple synchronous recordings and export them in multimodal animation files. AnimIO can be used to isolate motion segments that refer to individual craft actions, as described by practitioners. The proposed approach has been iteratively designed for use by non-experts in the domain of 3D motion digitisation.

Keywords: motion capture; computer vision; motion segmentation; motion retargeting; traditional crafts; virtual reality; cultural heritage; intangible cultural heritage

1. Introduction

Annotations upon recordings are statements that enrich the curated content, by classifying these recordings, or parts thereof, with respect to a vocabulary or by associating other media objects that provide further information; e.g., the identification of an event, or an explanatory narration of what is happening in the recorded scene. In traditional crafts and the production of handmade products, gestures are part of the production process and may be specific to a technique or the use of a tool. Photographic and video recordings were conventionally used to demonstrate actions. However, in this way, there is no measurement of motion that can serve documentation purposes and accurate re-enaction of the recorded actions. More recently, Motion Capture (MoCap) technologies enabled the acquisition of measurements that are essential in documenting human activity.

We call motion segmentation the task of separating regions, features, or trajectories from a motion recording that correspond to individual craft actions into coherent subsets of space and time. Motion segmentation is key in the comprehension of human activities because it enables the association of movements with entries in gesture vocabularies. By articulating motion into gestures, a more insightful understanding of an activity is achieved because individual gestures can be isolated, studied, and practiced. Using reference gestures, such as for rocking a treadle, sliding a shuttle, and pulling at the batter, weaving on a loom can be understood as the iteration of three

actions, rather than a continuous stream of hand and feet trajectories. In other words, motion segmentation requires action/gesture recognition. Pertinent methods have been proposed for visual methods and MoCap, e.g., [1–6].

What underlies recognition methods is the requirement for training examples. In the early days, model-based recognition was explored with an emphasis in well-defined gesture vocabularies, such as those of sign languages [7,8]. In these cases, gestures models were a priori specified and available in an analytical and unambiguous form. More recently, computer vision, parallel computing, and machine learning make the definition of gesture vocabularies based on examples more feasible [9,10]. The requirement of analytical gesture models has, thus, been reduced to the provision of unambiguous training data, meaning that (preferably several) examples of each vocabulary gesture are available.

In our case, the term training data means datasets of a person performing the activity of interest, annotated per semantic action we wish to reference. In this context, training datasets have been developed [11,12] that offer recordings of individual gestures, where an actor performs the gestures in the environment of a motion capture laboratory. The segmentation of the examples is provided "by construction" as training examples contain a gesture in each one. Existing datasets refer to generic gestures and activities (i.e., the CMU Graphics Lab Motion Capture Database http://mocap.cs.cmu.edu/), but no datasets for craft practice are available.

This work proposes a workflow for the creation of multimodal datasets. We consider that the proposed approach will be used from professionals, not the technological domain but the cultural heritage domain, whether they are curators or practitioners. As such, the proposed approach has been designed with simplicity and using known user-interface metaphors, borrowed from video playback. The objective of this design approach was to ensure that the outcome will be usable by all targeted stakeholders requiring minimum prior knowledge and domain expertise. The approach is validated in the context of motion recordings from two activities involved in traditional crafts, namely weaving and mastic cultivation.

2. Background and Related Work

2.1. Multimodal Annotation Editors

Multimodal annotation editors have emerged from computational linguistics, in studies related to human communication and interaction, i.e., [13–16]. The requirements included the annotation of multimodal recordings capturing primarily audio, but also gestures, postures, gaze, facial expressions and other factors.

The most prominent packages are Transana [17] and EXMARaLDA [18]. Although video is always the second modality after audio recording, these editors are mainly dialogue-oriented. The VISTA editor extends annotation capabilities by incorporating person location data, in order to monitor social communication and motion in wider spaces. The ELAN [19] video annotation editor is a tool for the annotation of interviews in controlled settings, which emphasises the simultaneous annotation of audio and video.

The most relevant work to the proposed one is the ANVIL [20] annotation editor, which offers facilities for the synchronisation and visualisation of MoCap data. A comparative evaluation of these applications can be found in [21]. Another tool used in the industry, as a 3D character animation software for the post-processing of animations, is MotionBuilder (https://www.autodesk.com/products/motionbuilder/overview), but this does not support the multimodal motion segmentation.

2.2. Motion Recordings

Motion Capture (MoCap) refers to a wearable human motion recording modality that acquires 3D measurements of human motion, with a variety of technologies having been used for its implementation [22]. The output of MoCap is a set of series of 3D locations. Each series records the

3D locations of a physical point of the body. Optical, or Computer Vision systems estimate the same information through 2D measurements in images or video.

The most widely used approach is optical MoCap and is achieved by the placement of markers on the moving body [23]. Off the shelf products include OptiTrack (https://optitrack.com) and Vicon (https://www.vicon.com/). The Inertial MoCap approach is based on Inertial Measurement Units (IMUs) to record motion, woven within an elastic suit [24]. Off-the-shelf products include NANSENSE Biomed MoCap System (https://www.nansense.com/) and Rokoko (https://www.rokoko.com). The topological mapping of markers follows that of the human skeleton. Marker-less optical systems infer 3D motion from conventional videos, using machine learning approaches that identify the human body and recognize its posture given appropriate training data [25,26].

Motion recordings vary in accuracy and precision depending on the MoCap acquisition technology [27]. In general, optical MoCap is more accurate than inertial MoCap; however, research on inertial tracking is active and producing frequent advances, tailored for the human body [28]. The least accuracy is provided by marker-less optical systems, due to a lack of direct 3D measurements. Post-processing of the recording is often required, a task that requires human supervision and interaction, and is facilitated by a range of visual tools, such as MotionBuilder (https://www.autodesk.com/products/motionbuilder/overview), Blender (https://www.blender.org/), and Maya (https://de.rebusfarm.net/en/3d-software/maya-render-farm), etc.

In all cases, the topology of markers is known and employed in the organisation of measurements. When capturing motion, markers are purposefully placed in association with the actual skeletal joints, delimiting its rigid components (actual bones). In this way, the hierarchic structure of the human skeleton is accounted for in the representation of human motion. In facial MoCap the markers are placed in association with anatomical locations on the face, such as the mouth and eyebrows, which are controlled by facial muscles and determine the expression of facial gestures (facial expressions).

2.3. This Work

Taking into consideration the above-mentioned related work and limitations, in this work, the following are provided.

1. An approach for the transformation of human motion recordings into VR demonstrations through segmentation of motion recordings (e.g., video, MoCap data, Visual Tracking outputs, etc.) and motion retargeting to Virtual Humans. This approach includes a solution to the interoperability problem of the multiple types of human motion recordings, as per the skeletal model.
2. A software application implementing the proposed approach named Animation Studio (AnimIO). The implementation, targets the widely adopted Unity3D (https://unity.com/) game engine, thus allowing wide adoption and further improvement of the presented research outcomes.
3. A simple user interface**.** The solutions described above require technical background on the topics of 3D surface representation, kinematics, and differential geometry. In this work, we strive to provide simple and straightforward solutions including tools to assist these tasks. AnimIO is iteratively designed and evaluation with the participation of usability and motion digitisation experts. AnimIO is provided as a resource tailored for use by non-experts in motion digitisation (non-domain experts).
4. Advanced motion visualisations. Three-dimensional visualisation of the recorded motion is provided by skeletal and avatar animation.
5. Photorealistic 3D visualisation based on motion retargeting, for the inclusion of results in VR demonstrations.

3. Overview of Approach

The proposed approach is overviewed in Figure 1. Our approach starts with understanding the studied craft process, defining pertinent activities, and recording them using video and/or MoCap. Afterwards, AnimIO is used to segment the recording and produce animation files and libraries of animation files. These libraries are imported in Unity3D and retargeted to avatars to be utilized within VR presentations.

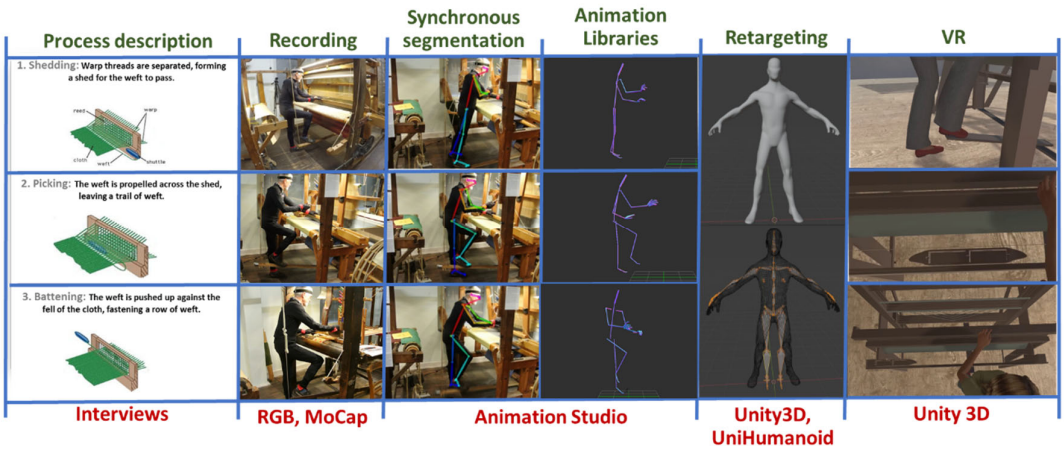

Figure 1. Overview of the proposed approach.

Process description: we start with interviews with the practitioners that will provide the demonstrations. This preparation is required to appropriately organize the acquisition of data, to understand the content and context of the recordings, and to provide instructions to practitioners regarding the recording protocol. Practitioners can be asked to explain the process and each task, to facilitate the annotation and motion segmentation of the data afterwards.

Recording: In this phase, practitioners are recorded by (a) one or more video cameras imaging the scene from multiple viewpoints and/or (b) a motion capture suit. Depending on the number and type of recording modalities, recordings are synchronised and consolidated. In the case of multiple videos, multiple sources are consolidated via a multi-view Computer Vision algorithm; in our case [25]. We do not deal with the case of consolidating input from visual tracking and MoCap modalities, as MoCap is always more accurate. The synchronisation between multiple video sources is treated by vendor-dependent camera synchronisation utilities. A special case of synchronisation is that between a video and a MoCap recording of the same event. Even though only the MoCap recording is utilised in the 3D animation output, video facilitates understanding of MoCap data by non-domain experts. Currently, this is achieved by external utilities and comprises a goal of future work.

Motion segmentation. The recordings are pre-processed for synchronization and AnimIO is operated to create multimodal motion segments. These segments contain synchronous recordings from multiple modalities and, in particular, video streams from multiple viewpoints or MoCap recordings. This is the phase where AnimIO is designed to be utilized by curators to encode their knowledge. This encoding refers to the annotation of motion data per activity executed. As such, AnimIO is designed to facilitate the temporal delimitation of motion segments by individual human limbs, while respecting constraints stemming from their hierarchical configuration.

Animation libraries: An animation library is a collection of animation files. Animation files are motion segments, typically containing the recording on an individual action by one or multiple recording modalities. The motion data are encoded in the skeletal model configuration. For each joint, a 3D location and a 3D rotation are encoded as an individual stream. The definition of the skeletal hierarchy is determined in the header of the animation file. Motion data can be previewed within AnimIO in two different forms: (a) skeletal animation and (b) simplified skeletal animation. It is important to note that in AnimIO 3D viewer, motion data can be previewed in different angles, scale, and locations, thus taking full advantage of the 3D information for close inspection.

Motion Retargeting: This maps 3D motion data from the animation file to a rigged avatar model in Unity3D and ensures that all aspects of the animation are appropriately transferred to the avatar through a bone-mapping procedure.

VR preview: This is the integration of the created animations to a VR environment, to preview the recorded craft processes performed by an avatar.

4. The AnimIO Video Annotation Editor

4.1. Rationale of Implementation

AnimIO was developed for the post-processing of multimodal recording of practitioner motions. AnimIO enables the visualisation, editing, and annotation of 3D animation files, obtained by Motion Capture (MoCap) or optical tracking. In this tool, the user simultaneously accesses multiple synchronised recordings and facilitates the isolation of multimodal recording segments. AnimIO was created to address the lack of a solution, which enables the annotation of segments, in synchronised video recordings and MoCap animations.

Furthermore, this work proposes an approach for simple and straightforward motion retargeting in Unity3D rather than requiring the usage of an intermediate application, i.e., Blender (https://www.blender.org/). Using Unity3D for the retargeting tasks directly simplifies the conventional workflow for this task.

4.2. Representation of Human Motion and Posture

Human posture is represented by a hypothetical skeleton. This skeleton is comprised of bones, each one representing a rigidly moving segment of the skeleton. Bones are hierarchically arranged. Human motion is represented by the animation of this skeleton in time.

Many craft actions do not regard the entire body but are relevant only to hand or feet, which are represented as branches in the hierarchical structure. Hierarchical representation of motion enables the annotation to refer to such an anatomical branch of the hierarchical structure. The representation of motion supported by this work acknowledges the hierarchical structure of the human skeleton and utilizes this concept to more specifically annotate the recorded motion.

For this purpose, AnimIO uses the BioVision Hierarchical data (BVH) format, an open, text-based format that supports a hierarchical structure that can be used to represent the human skeleton. Format specification details can be found in [29]. The BVH file consists of two parts. The first part of the BVH file defines the number of bones and their lengths, as well as their hierarchical organization. The number of bones is determined for inertial MoCap by the number of sensors and optical methods by the complexity of the skeleton model. The hierarchical organization defines the joints between the bones of the skeleton. The second part describes the recorded motion of this structure in time, as the modulation of joint angles. For each joint, three angles define a rotation in SO(3), thus enabling composition operations across branches of the hierarchy. This means that when a shoulder joint exhibits angular motion, the entire arm inherits this motion, including the elbow, wrist, and fingers. At the same time, individual angles at the joints of the latter are composed "on top" of the shoulder motion.

4.3. User Interface (UI)

The AnimIO UI is composed of four (4) main panels (Figure 2). The top panel (green) is the menu bar from which the users have access to the main project-based functionality. The middle panel is divided into two segments. The left segment (blue) is the video window, where the video is shown and included the video playback control buttons (play/pause/stop). The segment on the right (magenta) shows the 3D animation along with a toolbar for manipulating the observation viewpoint (translation/rotation/scale). The bottom panel (yellow) is the timeline which (i) controls both video and animation playback and (ii) enables the annotation of multimodal segments to be isolated within an animation file.

Video playback is controlled through the play/pause and stop buttons that are displayed at the bottom side of the video window. Both displays are synchronised, which means that when the video plays, the MoCap animation does so too.

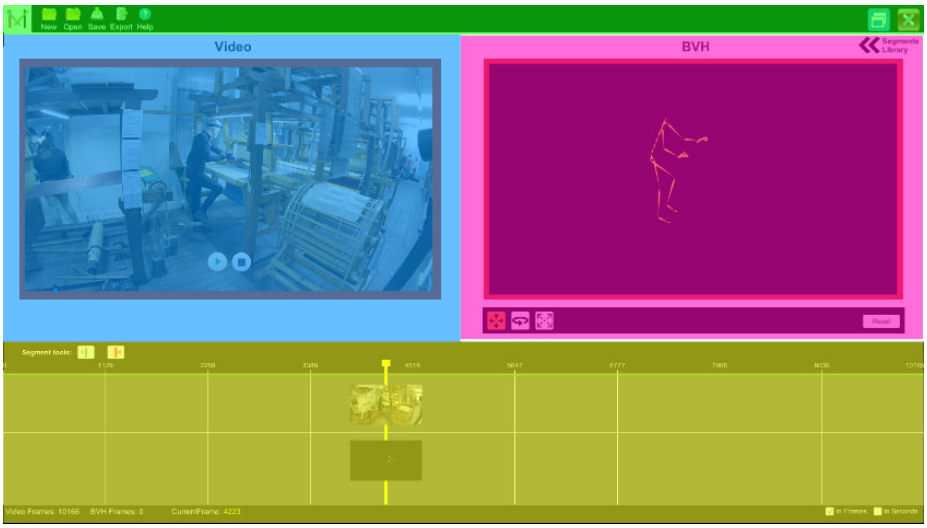

Figure 2. Editor window.

4.4. Multimodal Visualisation

AnimIO provides visualisation of the treated motion data in two forms: (a) skeletal visualisation and (b) avatar visualisation, for the following reasons. Skeletal visualisation is better for visualising row limp data while simplified visualisation is better for visualising motion for non-domain experts.

Skeletal visualisation regards the visualisation of the "raw" recorded 3D data and their skeletal relationships. A 3D point representation of joints and 3D line representation of bones provides a clear and one-to-one visualisation of the recorded data, as coordinates in space.

Avatar visualization presents a simple avatar that is appropriate for presenting the recorded motion to non-domain experts. A photo-realistic representation is provided later in the proposed approach through integration with the Unity3D game engine.

A design challenge was to support both 2D video and 3D animation, in a meaningful and intuitive fashion. The design choice made was to present video and 3D animation side by side (Figure 3). In some cases, the operator may need to rotate the scene to more conveniently observe the recorded motion or scale it for more detailed observation. This is supported through the 3D panel using the embedded controls (move, rotate and scale).

Figure 3. Video and 3D motion windows.

4.5. Motion Segmentation and Export

Motion segmentation is accomplished through the timeline panel. The current frame cursor (yellow vertical line) can be moved along the timeline and always sets the video and MoCap animation to the corresponding frame. Two thumbnail images are used on the cursor (Figure 4) to indicate the current video and animation frame. Furthermore, users can navigate the video and animation frame by frame by dragging the yellow cursor (yellow cursor always indicates the current frame) along the timeline. While doing so, the current frame of the video and the MoCap animation is displayed on the cursor.

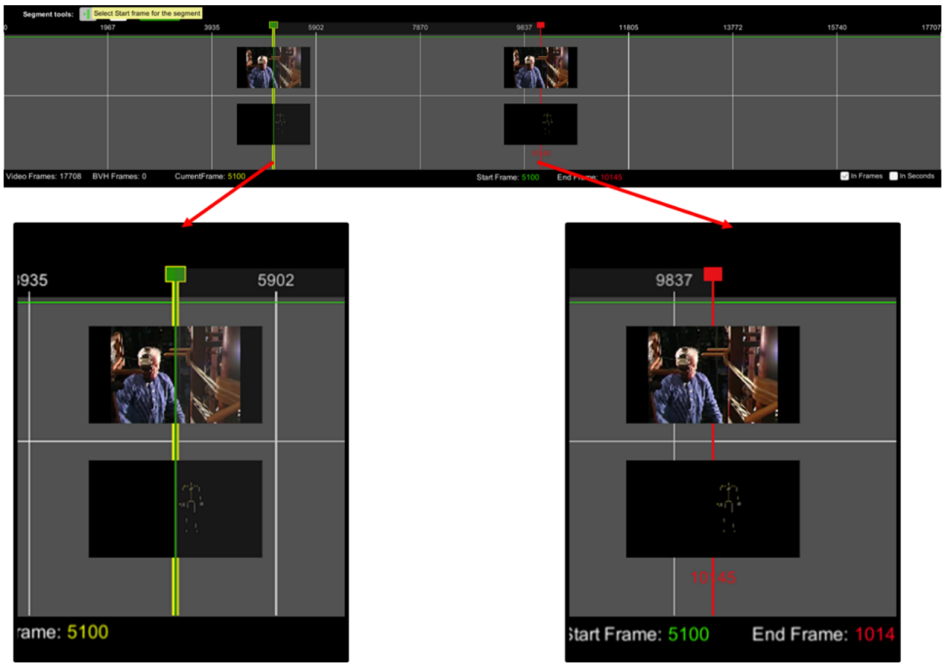

Figure 4. Timeline component (visualisation of start and end video and skeleton frame).

The timeline enables the creation of animation segments, by specifying its starting and ending frame in the timeline bar, using the start and end image icons (Figure 5, top left, green and red image button). Created segments are added to the library (Figure 5, right), which enables playing, editing, adding or removing individual segments. Segments can be also rearranged within the list.

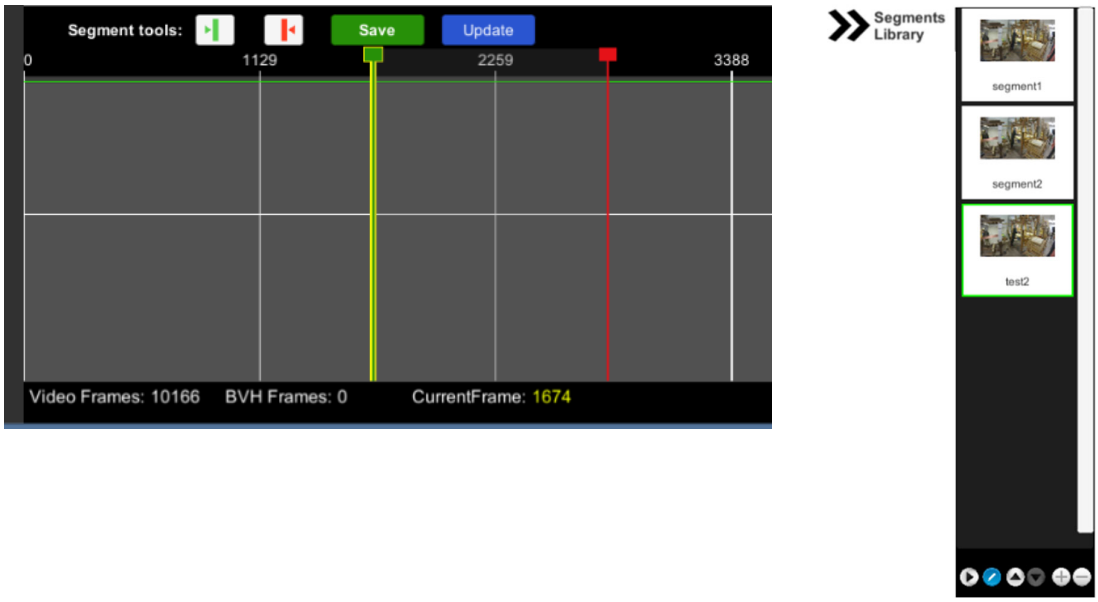

Figure 5. (**Left**): Timeline segment and controls. (Right): Segments library.

Users can export their created video and MoCap segments. Segments can be exported atomically or merged into a single video and animation file (Figure 6).

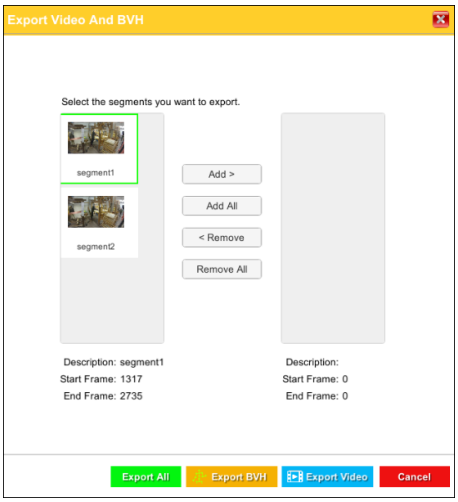

Figure 6. Export dialog.

4.6. Motion Retargeting

After exporting Animation Libraries from AnimIO, the next steps are to integrate them with the game engine and to retarget motion to avatars. To this end, the Unity3D game engine and the UniHumanoid (https://github.com/ousttrue/UniHumanoid) framework are used. UniHumanoid is a Unity3D framework for importing BVH files that encode human motion, by a skeletal model. Retargeting involves (a) importing a BVH file into Unity3D using the "BVH runtime loader", (b) importing the rigged character model, (c) bone-mapping for both rigged models (the one imported by the BVH and the one provided by the character) and (d) pose transfer from the rig described by the BVH file to the rig provided by the avatar. The result is a realistic animation performed by an avatar.

4.7. Implementation

4.7.1. Project Structure

Each project in AnimIO is stored in an individual file, which contains links to the input media sources. The structure of the motion files determines the profile that should be used to achieve the appropriate parsing and interpretations. Project files include additional parameters such as the number of animation frames and the segments the user has created. Each segment is defined by an id, a description, a start frame and an end frame, as well as a thumbnail, which is usually the first frame image of the selected part of the video or BVH. The format used for the project file is JSON (https://www.json.org/json-en.html) encoded strings.

4.7.2. BVH Parser

A BVH parser was created to load the BVH file and transform the data into data structures for usage in Unity3D, to visualize the skeleton and motion data. The BVH format does not include information about the scale and orientation of joints. As such, there exist multiple conventions utilized to encode motion, adopted by individual applications suites. AnimIO accounts for multiple BVH sources and conventions using Convention Profiles. These profiles allow conversion between left/right-handed systems and/or the definition of global or per-axis scaling factors. Furthermore, the profiles contain a sequence of the transformations required for the data to (a) be displayed appropriately in AnimIO and (b) be exported uniformly to simplify the retargeting task later on.

4.7.3. Video Integration

For video decoding, AnimIO employs FFMpeg (https://ffmpeg.org/), a powerful cross-platform multimedia framework that allows the manipulation of almost every video format available. FFMpeg is a command-line tool, thus we developed a Wrapper class that handles the processes required by the application. The Wrapper class encapsulates a thread responsible for executing the FFMpeg tool with the appropriate command-line arguments, receives the produced output, and provides it to the application. The duration of some of the invoked operations is long (i.e., merging videos) and, thus, we use a thread upon their usage to avoid compromising user experience.

5. Design

Rapid prototyping and agile development techniques were used. As such, in the first design iteration, paper-based prototypes were produced. These prototypes were used to implement the first functional version of the tool. The first version was inspected by a technology domain expert with experience in video editing software and techniques. This iteration produced a list of found issues and suggestions for improvements. The improvements were directly applied to the tool leading to its second version, which was then released for actual use for the post-processing of the collected motion recording datasets. Feedback received in the form of issues observed during its usage ('beta' testing) and suggestions for improvements led to the third design iteration of the tool. The third version of the tool was then evaluated by two human–computer interaction usability and interaction design experts, using the cognitive walkthrough inspection method. Reported issues and suggestions for improvements from this iteration were implemented in the tool with the close collaboration of an interaction designer and the development team. Based on the suggestions, the fourth version of the tool, incorporating all the above-mentioned suggestions, was implemented. The evaluation was part of the Mingei Innovation Action, of the H2020 Programme of the EC (http://www.mingei-project.eu/). AnimIO will be provided free of charge as an outcome of Mingei, as a tool for craft representation and presentation.

5.1. First Design Iteration

In the first design iteration, low fidelity prototyping in the form of paper sketches was produced to depict the tool's main UI structure (Figure 7 left). These prototypes were then used to implement the first functional UI prototype of the tool (Figure 7 right) using Unity 3D.

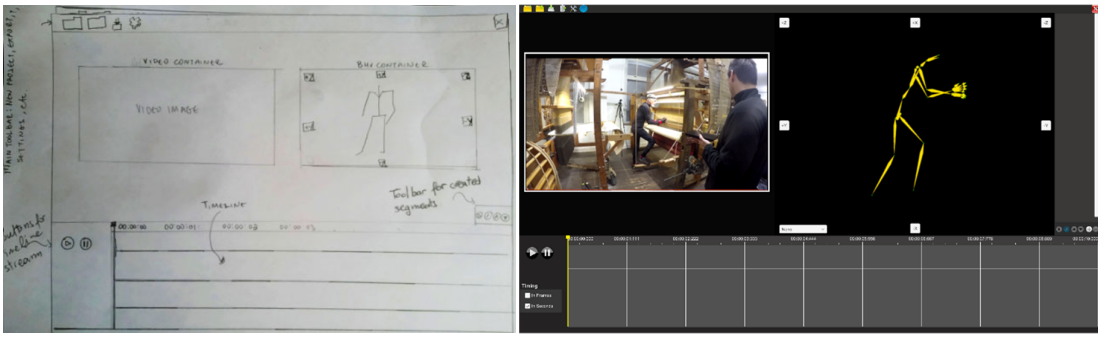

Figure 7. (Left): Paper prototype for AnimIO, (Right) AnimIO version 1.0.

The first inspection of the functional prototype conducted by a domain expert and interaction reported a list of issues along with suggestions for improvements. The expert found it difficult to manipulate the 3D preview of the motion due to the complex navigation structure and they suggested implementing direct mouse-based manipulation. Furthermore, in the timeline, the expert requested that we include image-based previews from the video and motion stream and simplify the selection of the start and end locations to perform a motion segmentation task. Additionally, the expert found that the video playback functionality was misplaced and should be below the video reproduction pane. Finally, a minor comment included the placement of labels for all image button controls and the support of both a full-screen and a windowed mode of operation.

5.2. Second Design Iteration

Based on the feedback received from the first inspection, adjustments in the design and the functionality were implemented on the functional application prototype (Figure 8).

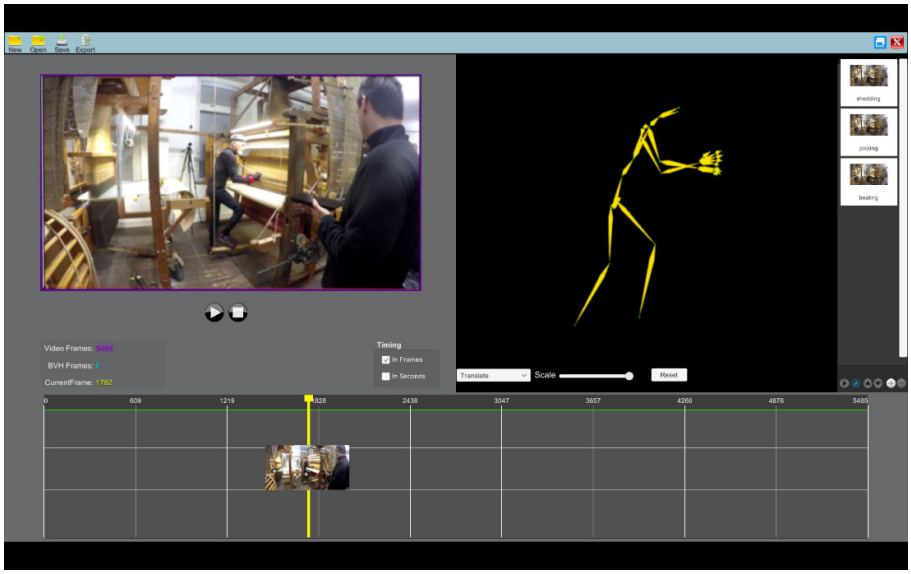

Figure 8. AnimIO version 2.0.

The second version of the tool was then distributed to the consortium partners of Mingei performing the motion segmentation task. During this iteration, consortium partners commented on the UI identity of the prototype and requested that we follow the identity of the project. Furthermore, they requested that we move the controls related to the timeline below the timeline strip to provide better mapping of functionality. Finally, they noticed that the video and 3D preview panels should have the same size and requested that we make the segments list foldable so as not to be constantly present while using the software. The combined feedback from the above users led to another round of design improvements implemented on the functional tool.

5.3. Third Design Iteration

Based on the feedback received from the second inspection, adjustments in the design and the functionality were implemented on the functional application prototype. As these were minor adjustments, the improvements were implemented directly on the functional version for quick release.

The resultant, improved version of the functional application prototype (Figure 9) was then inspected by two HCI usability and interaction design experts, who conducted a cognitive walkthrough. As these experts had no prior familiarity with the tool and its purpose, a description of the context of use and its purpose was given to them before the evaluation. The outcomes of the cognitive walkthrough evaluation resulted in aesthetic, functional, and usability recommendations and some additional general comments.

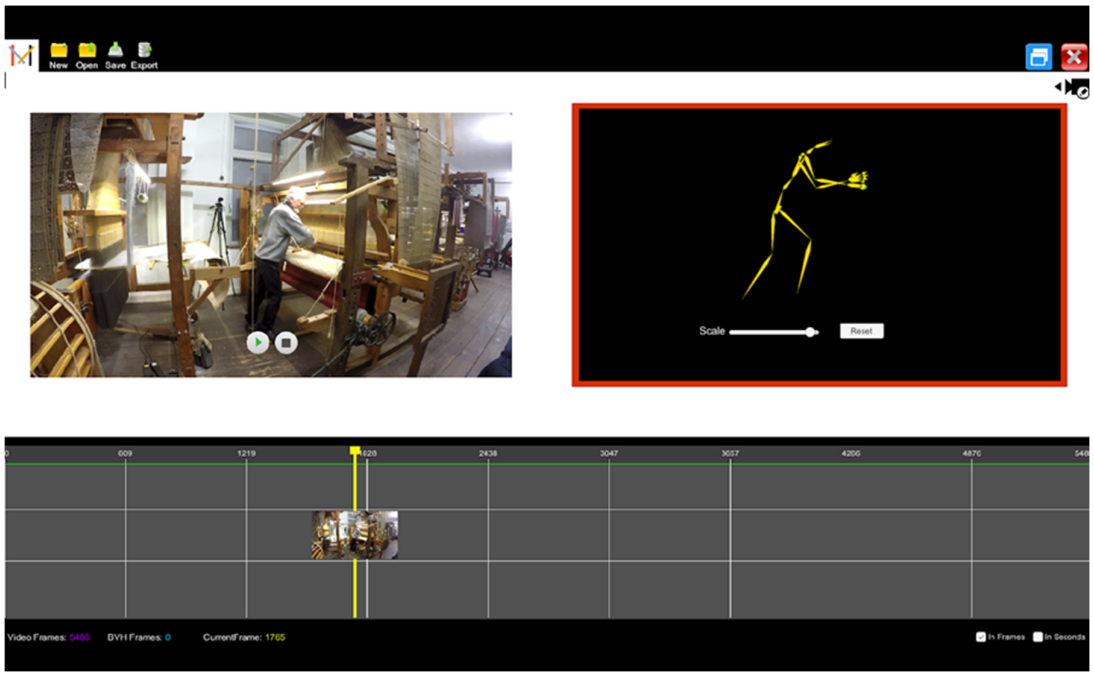

Figure 9. AnimIO, version 3.0.

Aesthetic recommendations included the enhancement of the eligibility of UI fonts in high resolutions, the addition of titles on the video and 3D display containers and the selection of more readable icons for the expand and collapse button for the library of segments container.

Functional recommendations included the support for importing video files after the creation of a project, the inclusion of buttons for complex 3D operations such as scale and rotate, the addition of frames from the 3D motion stream in the timeline preview, the provision of a toolbar for users to select the start and end frames of a segment and the updating of the status of playback buttons when clicked (on the segments library) following the same style as used in the video playback panel.

The usability recommendations included the addition of descriptive hover labels for all button and the provision of a help page with instructions on how to use the tool.

Overall the general comments of the evaluators were that in its final form, the software tool will be easy to use and learn as it uses a conventional UI structure and elements found in common video editing software. Even a user with average computer skills and some limited experience in editing would be able to use it effectively, as the complex operations are hidden from the user. The tool is easy to install and open in a desktop computer, as it requires a single click of the initialization button. Adding a help section in the tool would ensure that a beginner user would get the appropriate guidance in using this tool.

5.4. Fourth Design Iteration

The fourth version of the tool incorporated all the above-mentioned suggestions (see Figure 10). Major improvements include the timeline visualisation (see Figure 10, timeline container) and the preview of animation files via on-the-fly retargeting of animation to avatars (see Figure 11). This produces a more readable representation and integrates the same approach that we follow for retargeting motion in Unity3D (see Section 6.4.2.).

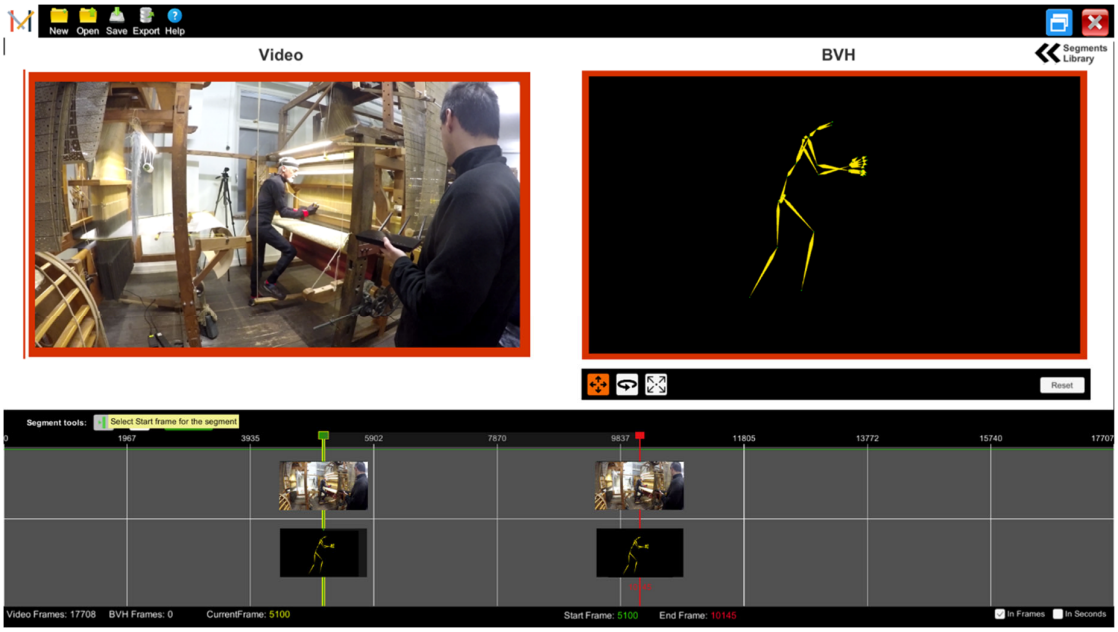

Figure 10. Screenshot of AnimIO version 4.0.

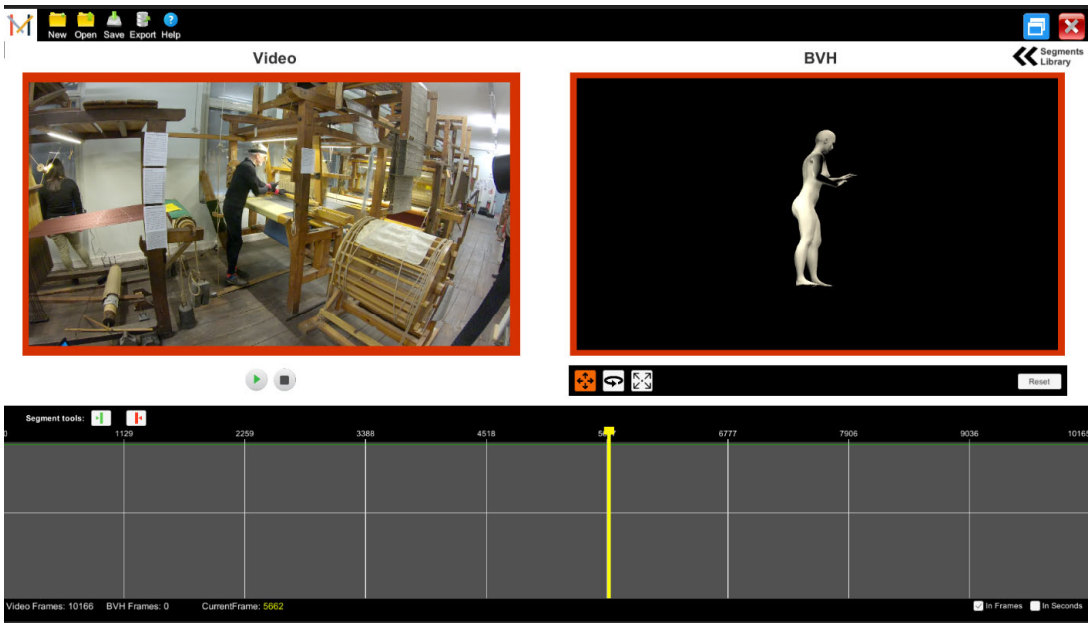

Figure 11. Previewing animation files by retargeting motion to avatars.

6. Use Case

This section presents usage examples of the proposed approach, in the context of textile weaving practiced.

6.1. Process Description

Interviews and workshops with craft practitioners were organised with the purpose of understanding craft processes. The interviews were structured based on an interview plan that was the result of our background study. Interviews were task-oriented. This study was conducted using online digital resources, publications, and archives. During the interview, the facilitators took notes regarding the sequences of craft processes to be recorded. The outcomes of these interviews were encoded in notes and illustrations of the sequence of actions together with a vocabulary of terms to assist the recording sessions. The organisation of workshops with the participation of multiple practitioners provided the dynamics to collaboratively work on craft understanding. Interaction between practitioners revealed details as well as craft process aspects that are considered important by practitioners and sometimes not evident to an observer without prior knowledge. The results enabled the planning of the recording activities and acquisition of data.

6.2. Motion Recordings

The recording sessions were organised by studying the loom area and organising the placement of recording equipment. The practitioner was recorded using inertial MoCap and marker-less optical tracking from video.

The practitioner motion was captured using the NANSENSE system. This system consists of a full-body suit composed of 52 IMU sensors placed throughout the body and hands. The sensors allow measuring the articulated spine chain, shoulders, limbs, and fingertips at a rate of 90 Hz.

Video recordings of the session were acquired from two views. The objective was to support data segmentation. In this way, each gesture is described by one motion file and two video segments from the respective viewpoints. Given the tasks planned to be recorded, camera placement ensured that the recording contained the views required for the sufficient documentation of the task. For example, in the case of weaving, we had to ensure a viewpoint from which the hands of the practitioner were imaged without occlusions by the loom apparatus. Furthermore, auxiliary

recordings were needed to assist segmentation tasks. In these recordings, the practitioner was requested to describe the process before execution to facilitate the annotation and segmentation of the data afterwards. Apart from the documentation of gestures, visual recordings facilitated the segmentation task as they are easily comprehensible in comparison to visualisation of only MoCap data. An example of the recording setup for a large-scale loom is presented in Figure 12.

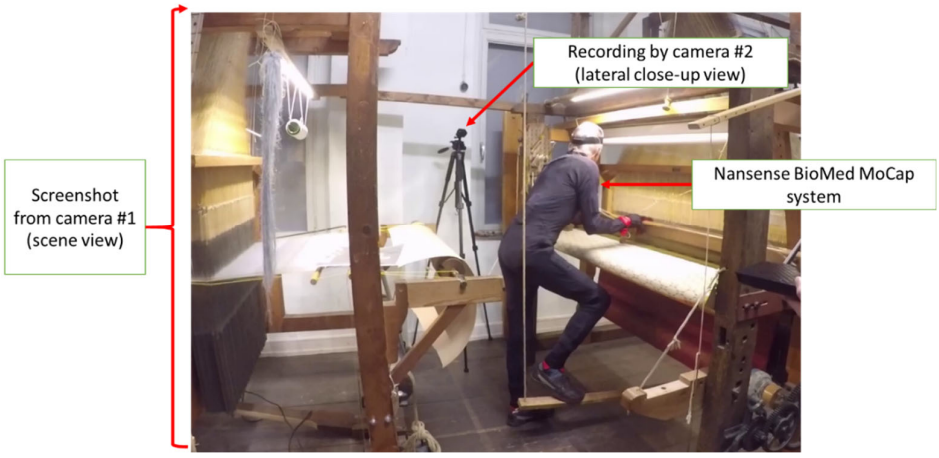

Figure 12. Motion recording setup.

6.3. Motion Segmentation

The recordings contain several useless segments that include technical preparation, mistakes, or idle time. To cope with this situation, an initial segmentation of recordings into "scenes" took place, where only meaningful data were kept for further processing. Scenes corresponded to craft activities. For each activity, a set of gestures was segmented. Furthermore, key poses for the identified gestures were extracted from the video. The result of this process was the implementation of a vocabulary of gestures that contains (a) video segments, (b) key poses, (c) motion segments, (d) and motion segments from visual tracking. For the loom weaving activity, the segmentation task resulted in a gesture vocabulary consisting of 16 gestures, as shown in Table 1.

Table 1. Multimodal gestures vocabulary for weaving.

Task	Number of Gestures
Creating a card	1
Beam preparation	5
Wrapping the beam	1
Weaving with small size loom	3
Weaving with medium size loom	3
Weaving with large size loom	3

6.4. Animation Libraries and Motion Retargeting

6.4.1. Implementation of Avatars

The compatibility of the approach with several avatars exported from 3D modelling software was tested. For this purpose, multiple avatars were created in Poser Pro 11 (https://www.posersoftware.com/), Adobe Fuse (https://www.adobe.com/gr_en/products/fuse.html) and Character Creator (https://www.reallusion.com/character-creator/). These avatars were exported from the creation software and imported to Unity3D. Then, the retargeting process was done following the methodology presented in the next section. The only variation of the procedure was

the bone mapping process which depends on the skeletal structure of the imported avatar. In this way, we were able to validate the correct retargeting of motion across a wide range of body types. We conclude that the methodology was replicable with successful mapping of motion for all tested avatars.

6.4.2. Motion Retargeting

For the retargeting task, the animation libraries were imported to Unity3D in the form of motion data in BVH format. The retargeting task involves mapping the bones of an "ideal" skeletal structure to the bones of the imported avatar. This is achieved by the usage of the UniHumanoid library. Pose transfer is used to make the avatar assume the posture of the ideal skeletal structure. The result of this task is the transformation of motion data to an animation performed by the avatar. Figure 13 shows an example for the case of weaving acquired by the usage of the segmented data of the loom (see Table 1) in the recording context presented by Figure 12. The main advantage of this method is that motion is retargeted gracefully without artefacts that result in unrealistic animations.

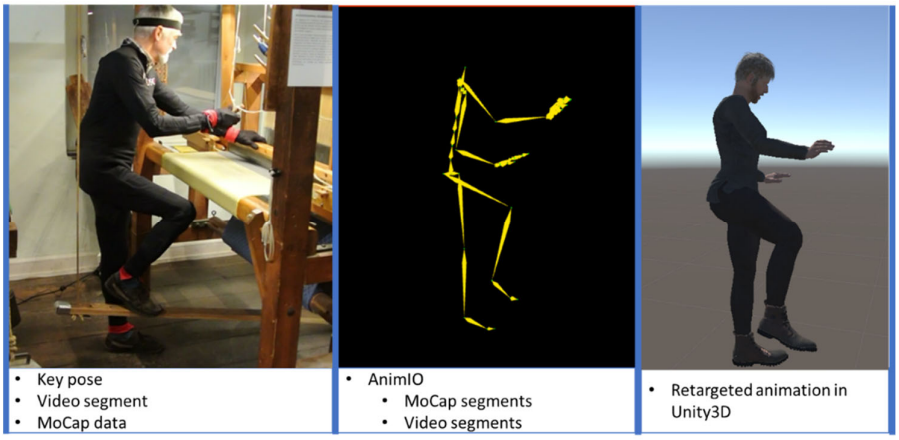

Figure 13. Weaving motion recording using NANSENSE system (left), segmented MoCap in AnimIO (middle), retargeted animation in Unity3D (right).

6.5. VR Preview

For previewing the animations in a 3D virtual environment, the following method was followed. Initially, a 3D model of a building structure was used to create the virtual world where the crafting process will be presented (https://assetstore.unity.com/packages/3d/environments/urban/furnished-cabin-71426). A loom model was imported into the scene (https://3dwarehouse.sketchup.com/model/aefc831b85e940fac2f4a75b7b3fb6d0/Loom-szovoszek) and the avatar was placed in his initialisation state (T-pose) in approximation with the loom. For the demonstration, an animation sequence was generated that iterates the basic weaving animations. In Figure 14, a screen capture from the weaving processes in VR is shown.

Figure 14. Weaving VR example.

6.6. Lessons Learned

The presented approach is the result of several experiments with tools and technologies conducted in the context of the Mingei project. In this project, the need for a motion segmentation pipeline was quickly realised. Although this was possible in other software platforms, there was no way to perform segmentation of motion and video sources concurrently. This was a prerequisite from our side to support documentation of craft processes in multiple media, but also it was needed to facilitate segmentation through the presentation of visual input to operators. Visual input simplifies the processing of motion data that are previewed usually, by a simplified stick figure animation. AnimIO was an outcome of experimentation with existing technologies and the resulting lack of a robust software solution to achieve the aforementioned goals.

Other issues were also encountered. During trial and error experimentation, unpredictable results occurred in motion retargeting. Some of these include incomplete motion transfer, skeletal model inconsistencies, scaling issues, etc. For example, initially, integration of motion and the creation of animations was conducted in external software and then the animations were exported together with the avatar and imported to Unity3D. This resulted in several incompatibility issues and animation problems; e.g., animation is previewed in the source software correctly but not in the destinations, etc. Thus, a reusable methodology could enable this task to take place in a more organised and predictable fashion. UniHumanoid simplified bone-mapping and motion retargeting, as both are performed in a single software environment and with repeatable results.

The selection of Unity3D, a widely adopted platform, made it possible to accomplish mapping, retargeting, and a preview of motion both in 3D and VR using the same software stack, thus simplifying the application of the proposed methodology.

7. Discussion and Future Work

In this work, an approach to the transformation of human motion to VR demonstrations was presented. Regarding future work interventions, the upcoming advancements include isolating motion of certain body members. This is useful when (a) presenting a gesture that involves only some of the limbs of the body and when (b) presenting concurrent body motions. To "silence" the motion of some limbs hierarchically, we are going to employ the rigged model hierarchy. This follows much of the approach on human motion representation, formed in [30]. By allowing the selection of different simplified animation elements, such as cubes and spheres, AnimIO is intended to simplify

this task. Furthermore, future improvements include the simultaneous handling of multiple video sources. Finally, to further elaborate on the usability of the tool, a user-based study is planned to validate system design and evaluate usability and user experience for different user types (e.g., researchers, curators, users from the Cultural and Creative sector).

Author Contributions: For research articles with several authors, a short paragraph specifying their individual contributions must be provided. The following statements should be used "conceptualization, N.P. and X.Z.; methodology, N.P and X.Z.; software, N.P, N.P, A.C.; validation, I.A., N.P. and X.Z.; writing—original draft preparation, N.P., X.Z and A.C.; writing—review and editing, N.P. and X.Z.; visualization, A.C. and N.P; supervision, X.Z.; project administration, X.Z.; funding acquisition, N.P and X.Z. All authors have read and agreed to the published version of the manuscript.", please turn to the CRediT taxonomy for the term explanation. Authorship must be limited to those who have contributed substantially to the work reported.

Funding: This research has been conducted in the context of the Mingei project that has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 822336.

Acknowledgements: This work has been conducted in the context of the Mingei project that has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 822336. MoCap recordings and segmentation results were acquired by the Association pour la Recherche et le Developpement des Methodes et Processus Industriels (ARMINES) in the context of their contribution in the project. Authors thank the Association of Friends of Haus der Seidenkultur for the weaving demonstrations and the provision of insights in understanding the craft of textile manufacturing.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Lan, R.; Sun, H. Automated human motion segmentation via motion regularities. *Vis. Comput.* **2015**, *31*, 35–53.
2. Schulz, S.; Woerner, A. Automatic motion segmentation for human motion synthesis. In *International Conference on Articulated Motion and Deformable Objects*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 182–191.
3. Kadu, H.; Kuo, C. Automatic human MoCap data classification. *IEEE Trans. Multimed.* **2014**, *16*, 2191–2202.
4. Li, C.; Kulkarni, P.; Prabhakaran, B. Segmentation and recognition of motion capture data stream by classification. *Multimed. Tools Appl.* **2007**, *35*, 55–70.
5. Müller, M.; Röder, T. Motion templates for automatic classification and retrieval of motion capture data. In Proceedings of the 2006 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, Vienna, Austria, 2–4 September 2006; pp. 137–146.
6. García-Vega, S.; Álvarez-Meza, A.; Castellanos-Domínguez, C. MoCap data segmentation and classification using kernel based multi-channel analysis. In *Iberoamerican Congress on Pattern Recognition*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 495–502.
7. Triesch, J.; Von Der Malsburg, C. A gesture interface for human-robot-interaction. In Proceedings of the Third IEEE International Conference on Automatic Face and Gesture Recognition, Nara, Japan, 14–16 April 1998; pp. 546–551.
8. Birk, H.; Moeslund, T.B.; Madsen, C.B. Real-time recognition of hand alphabet gestures using principal component analysis. *Proc. Scand. Conf. Image Anal.* **1997**, *1*, 261–268.
9. Barnachon, M.; Bouakaz, S.; Boufama, B.; Guillou, E. Ongoing human action recognition with motion capture. *Pattern Recognit.* **2014**, *47*, 238–247.
10. Kapsouras, I.; Nikolaidis, N. Action recognition on motion capture data using a dynemes and forward differences representation. *J. Vis. Commun. Image Represent.* **2014**, *25*, 1432–1445.
11. Singh, R.; Sonawane, A.; Srivastava, R. Recent evolution of modern datasets for human activity recognition: a deep survey. *Multimed. Syst.* **2020**, *26*, 83–106.
12. Chaquet, J.; Carmona, E.; Fernández-Caballero, A. A survey of video datasets for human action and activity recognition. *Comput. Vis. Image Underst.* **2013**, *117*, 633–659.
13. Hereld, M. SIDGrid: A Framework for Distributed, Integrated Multimodal Annotation, Archiving, and Analysis. Relation. 2008. Available online: https://faculty.washington.edu/levow/papers/sigdial07.pdf (accessed on 31 August 2020).

14. Levow, G.A.; Bertenthal, B.; Hereld, M.; Kenny, S.; McNeill, D.; Papka, M.; Waxmonsky, S. SIDGRID: A Framework for Distributed and Integrated Multimodal Annotation and Archiving and Analysis. In Proceedings of the SIGdial Workshop on Discourse and Dialogue, Antwerp, Belgium, 1–2 September 2007; pp. 231—234.
15. Blache, P.; Bertrand, R.; Bruno, E.; Bigi, B.; Espesser, R.; Ferré, G.; Guardiola, M.; Hirst, D.; Tan, N.; Cela, E.; et al. Multimodal Annotation of Conversational Data. In Proceedings of the Fourth Linguistic Annotation Workshop, Uppsala, Sweden, 15–16 July 2010; pp 186–191.
16. Ito, K.; Saito, H. An Annotation Tool for Multimodal Dialogue Corpora using Global Document Annotation. In Proceedings of the SIGDIAL Workshop, Sapporo, Japan, 5–6 July 2003; pp. 192–197.
17. Woods David, K.; Dempster Paul, G. Tales from the Bleeding Edge: The Qualitative Analysis of Complex Video Data Using Transana. In *Forum Qualitative Sozialforschung/Forum: Qualitative Social Research*; K. Mruck: Berlin, Germany, 2011; Volume 12, ISSN 1438-5627, doi:0.17169/fqs-12.1.1516
18. Schmidt, T.; Worner, K. EXMARaLDA—Creating, analysing and sharing spoken language corpora for pragmatic research. *Pragmatics* **2009**, *19*, doi:10.1075/prag.19.4.06sch.
19. ELAN. Available online: https://archive.mpi.nl/tla/elan (accessed on 31 August 2020).
20. Kipp, M.; von Hollen, L.; Hrstka, M.C.; Zamponi, F. Single-Person and Multi-Party 3D Visualizations for Nonverbal Communication Analysis. In Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC), ELDA, Paris, France, 26–31 May 2014.
21. Rohlfing, K.; Loehr, D.; Duncan, S.; Brown, A.; Franklin, A.; Kimbara, I.; Milde, J.T.; Parrill, F.; Rose, T.; Schmidt, T.; et al. Comparison of Multimodal Annotation tools: Workshop report. *Gesprachsforschung* **2006**, *7*, 99–123.
22. Field, M.; Stirling, D.; Naghdy, F.; Pan, Z. Motion capture in robotics review. In Proceedings of the 2009 IEEE International Conference on Control and Automation, Christchurch, New Zealand, 9–11 December 2009; pp. 1697–1702.
23. Yahya, M.; Shah, J.; Kadir, K.; Yusof, Z.; Khan, S.; Warsi, A. Motion capture sensing techniques used in human upper limb motion: a review. *Sens. Rev.* **2019**, *39*, 504–511.
24. Shi, G.; Wang, Y.; Li, S. Human Motion Capture System and its Sensor Analysis. *Sens. Transducers* **2014**, *172*, 206.
25. Qammaz, A.; Argyros, A.A. MocapNET: Ensemble of SNN Encoders for 3D Human Pose Estimation in RGB Images. In Proceedings of the British Machine Vision Conference (BMVC 2019), Cardiff, UK, 9–12 September 2019; p. 46.
26. Qammaz, A.; Michel, D.; Argyros, A. A hybrid method for 3d pose estimation of personalized human body models. In Proceedings of the 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), Lake Tahoe, NV, USA, 9 February 2018; pp. 456–465.
27. van der Kruk, E.; Reijne, M.M. Accuracy of human motion capture systems for sport applications; state-of-the-art review. *Eur. J. Sport Sci.* **2018**, *18*, 806–819.
28. Loper, M.; Mahmood, N.; Black, M.J. MoSh: Motion and shape capture from sparse markers. *ACM Trans. Graph. (TOG)* **2014**, *33*, 1–13.
29. Lander, J. *Working with Motion Capture File Formats*; Game Developer: Hong Kong, China, 1998.
30. Marr, D. *Vision*; W.H. Freeman: San Francisco, CA, USA, 1982; pp. 41–98.